

## AMENDMENTS TO THE CLAIMS

1 (Previously Presented) A method of providing an interface to a database, wherein  
2 the database includes a plurality of nonhierarchically organized classifiers of data  
3 and data linked to at least one of the classifiers, the method comprising:  
4 organizing a set of the plurality of nonhierarchically organized classifiers into a first  
5 hierarchical data structure according to a view established for a first client;  
6 presenting data to the first client according the first hierarchical data structure;  
7 organizing a second set of the plurality of classifiers into a second hierarchical data  
8 structure according to a view established for a second client; and  
9 presenting data to the second client according the second hierarchical data structure.

1 (Original) The method as recited in Claim 1, wherein the first hierarchical data  
2 structure is a file system directory tree structure.

1 (Original) The method as recited in Claim 1, wherein the set of the plurality of  
2 classifiers contains less than the plurality of classifiers.

1 (Original) The method as recited in Claim 1, wherein the first hierarchical data  
2 structure is a file system directory tree structure according to NFS.

1 (Previously Presented) A method of presenting data in a database, wherein the  
2 database includes nonhierarchically organized classifiers(1:M) of data and data  
3 linked to at least one of classifiers(1:M), wherein "M" is any positive number  
4 including 1, the method comprising:  
5 creating a view for a client, wherein the view organizes a set of the nonhierarchically  
6 organized classifiers (1:M) into a hierarchical data structure according to  
7 characteristics of the client;  
8 presenting data to the client according to the hierarchical data structure.

1 (Original) The method as recited in Claim 5, wherein the hierarchical data  
2 structure is a file system directory tree structure.

1 (Original) The method as recited in Claim 5, wherein the database includes N  
2 classifiers and N is a number greater than M.

1 (Original) The method as recited in Claim 5, wherein the hierarchical data  
2 structure is a file system directory tree structure according to NFS.

1 (Previously Presented) A method of presenting data in a database in a  
2 hierarchical data structure, wherein classifiers that classify the data are  
3 nonhierarchically organized, the method comprising:  
4 receiving a request from a client;  
5 translating the request into a database query for data classified by the nonhierarchically  
6 organized classifiers;  
7 producing a set of data according to the database query, the set of data organized in a  
8 hierarchical data structure; and  
9 presenting the set of data to the client.

1 (Original) The method as recited in Claim 9, wherein translating the request into  
2 a database query comprises:  
3 looking up the file handle in an ILocation table to obtain an ILocation; and  
4 formulating a database query to query a defined set of data, the query formulated from  
5 the ILocation.

1 (Original) The method as recited in Claim 10, wherein formulating a database  
2 query to query a defined set of data comprises:  
3 determining if the ILocation contains bound classifiers, each bound classifier being  
4 bound by a constraining value;  
5 if the ILocation contains bound classifiers, adding a clause to the database query for each  
6 bound classifier in the ILocation, each clause formulated to eliminate data from  
7 the defined set of data that has a defined value corresponding to the constraining  
8 value of the bound classifier;  
9 determining if the ILocation contains unbound classifiers;  
10 if the ILocation contains unbound classifiers, adding a first clause to the database query  
11 for the first unbound classifier, the clause formulated to produce a listing of  
12 distinct values set for the defined set of data, the distinct values corresponding to  
13 the first unbound classifier, and further, adding a second clause to the database  
14 query for the first unbound classifier, the second clause formulated to select data  
15 from the set of defined data that has the value of the first unbound classifier not  
16 set to a value; and  
17 if the ILocation does not contain unbound classifiers, adding a clause to the database  
18 query that selects all data in the defined set of data.

1 (Original) The method as recited in Claim 10, wherein formulating a database  
2 query to query a defined set of data includes formulating an SQL query,  
3 comprising:  
4 determining if the ILocation contains bound classifiers, each bound classifier being  
5 bound by a constraining value;  
6 if the ILocation contains bound classifiers, adding a WHERE clause with a condition to  
7 the SQL query for each bound classifier, wherein the condition of each WHERE  
8 clause is set to the constraining value corresponding to each bound classifier;  
9 determining if the ILocation contains unbound classifiers;  
10 if the ILocation contains unbound classifiers, adding a SELECT DISTINCT clause to the  
11 SQL query for the first unbound classifier and further adding a WHERE clause  
12 with a condition to the SQL query for the first unbound classifier, wherein the  
13 condition of the WHERE clause is set to NULL; and  
14 if the ILocation does not contain unbound classifiers, adding a SELECT clause to the  
15 SQL query.

1 (Original) The method as recited in Claim 9, wherein producing a result using the  
2 database query, the result formatted according to the NFS protocol, comprises:  
3 querying the database with the database query that produces ILocations and files;  
4 translating the ILocations into unique file handles; and  
5 storing the mapping from the unique file handles to the ILocations in an ILocation table.

1 (Original) The method as recited in Claim 13, wherein translating the Ilocations  
2 into unique file handles comprises:  
3 translating the ILocations into single byte arrays;  
4 cryptographically hashing the single byte arrays into small byte arrays; and  
5 padding the small byte arrays with bits to make the small byte arrays the proper length of  
6 a file handle according to the NFS protocol.

1 (Original) The method as recited in Claim 9, further comprising providing a file  
2 handle to a client upon an initial access request from the client, the file handle  
3 corresponding to a view in the database, wherein the view defines an amount of  
4 data in the database that is observable by the client.

1 (Original) The method as recited in Claim 15, wherein the view defines a  
2 directory structure observed by the client.

1 (Previously Presented) A data processing system comprising:  
2 a monitor;  
3 a processor coupled to the display; and  
4 a memory coupled to the processor, the memory including instructions to cause the  
5 processor to graphically display a hierarchy of data classification information,  
6 wherein the data classification information represents classifiers of data, the  
7 classifiers are nonhierarchically organized in at least one database, the data is  
8 contained in the at least one database, the data is at least a portion of all data  
9 contained in the at least one database, and each datum is classified by at least one  
10 nonhierarchically organized classifier.

1 (Currently Amended) The data processing system as in claim 17 wherein the  
2 hierarchy is selectable.

1 (Previously Presented) The data processing system as in claim 17 wherein the  
2 display is in the form of a directory tree-like display.

1 (Previously Presented) The data processing system as in claim 17 wherein the  
2 database is remotely disposed from a client system, the further comprising:  
3 a graphical display of the data classification information on a display of the client system.

1 (Previously Presented) The data processing system as in claim 17 further  
2 comprising:  
3 an active link associated with each classifier represented by the data classification  
4 information that allows display of the data that is associated with each classifier.

1 (Previously Presented) A method of providing hierarchical data classification  
2 information, the method comprising:  
3 receiving a request for hierarchical data classification information, wherein the data  
4 classification information represents classifiers of data, the classifiers are  
5 nonhierarchically organized in at least one database, the data is contained in the at  
6 least one database, the data is at least a portion of all data contained in the at least  
7 one database, and each datum is classified by at least one classifier;  
8 generating the hierarchical database classification information; and  
9 providing the hierarchical database classification information.

1 (Original) The method of providing hierarchical data classification information as  
2 in claim 22, wherein generating the hierarchical database classification  
3 information comprises:  
4 translating the request into a database query; and  
5 producing a set of data according to the database query, the set of data organized in a  
6 hierarchical data structure.

1 (Original) The method of providing hierarchical data classification information as  
2 in claim 23, wherein translating the request into a database query comprises:  
3 looking up a file handle in an ILocation table to obtain an ILocation; and  
4 formulating a database query to query a defined set of data, the query formulated from  
5 the ILocation.

1 (Original) The method of providing hierarchical data classification information as  
2 in claim 22, wherein providing the hierarchical database classification information  
3 further comprises:  
4 providing the hierarchical database classification information in the form of a directory  
5 tree-like structure.

1 (Original) The method of providing hierarchical data classification information as  
2 in claim 22, wherein providing the hierarchical database classification information  
3 further comprises:  
4 providing the hierarchical database classification information from a server system;  
5 the method further comprising:  
6 displaying the hierarchical database classification information with a client  
7 system.

1 (Previously Presented) A computer readable medium having stored thereon a  
2 hierarchical data structure of classifiers of a database, wherein the classifiers  
3 include nonhierarchically organized classifiers(1:M) of data and data linked to the  
4 at least one of the classifiers(1:M), wherein M is any positive number including 1,  
5 the hierarchical data structure generated by the method of:  
6 creating a view for a client, wherein the view organizes a set of the nonhierarchically  
7 organized classifiers(1:M) into a hierarchical data structure according to  
8 characteristics of the client; and  
9 organizing data into the hierarchical data structure according to the set of the  
10 classifiers(1:M).

1 (Previously Presented) The computer readable medium as recited in Claim 27,  
2 wherein the hierarchical data structure is a file system directory tree structure.

1 (Previously Presented) The computer readable medium as recited in Claim 27,  
2 wherein the database includes N classifiers and N is a number greater than M.

1 (Previously Presented) The computer readable medium as recited in Claim 27,  
2 wherein the hierarchical data structure is a file system directory tree structure  
3 according to NFS.

1 (Previously Presented) A computer readable medium for providing an interface  
2 to a database, wherein the database includes nonhierarchically organized  
3 classifiers(1:M) of data and data linked to at least one of classifiers(1:M), wherein  
4 "M" is any positive number including 1, the computer readable medium  
5 comprising a set of instructions for enabling a computer system to:  
6 organize a set of the nonhierarchically organized classifiers(1:M) into a first hierarchical  
7 data structure according to a view established for a first client;  
8 present data to the first client according the first hierarchical data structure;  
9 organize a second set of the nonhierarchically organized classifiers (1:M) into a second  
10 hierarchical data structure according to a view established for a second client; and  
11 present data to the second client according the second hierarchical data structure.

1 (Original) The computer readable medium as recited in Claim 31, wherein the  
2 first hierarchical data structure is a file system directory tree structure.

1 (Original) The computer readable medium as recited in Claim 31, wherein the  
2 database includes N classifiers and N is a number greater than M.

1 (Original) The computer readable medium as recited in Claim 31, wherein the  
2 hierarchical data structure is a file system directory tree structure according to  
3 NFS.

1 (Original) The computer readable medium of Claim 31 wherein the computer  
2 readable medium is selected from the group comprising: a hard disk drive, optical  
3 drive, floppy disk drive, compact disk, or electronic signals representing the  
4 instructions.

1 (Previously Presented) An apparatus for providing an interface to a database,  
2 wherein the database includes nonhierarchically organized classifiers(1:M) of data  
3 and data linked to at least one of the classifiers(1:M), wherein "M" is any positive  
4 number including 1, the apparatus comprising:  
5 means for organizing a set of the nonhierarchically organized classifiers(1:M) into a first  
6 hierarchical data structure according to a view established for a first client;

7 means for presenting data to the first client according the first hierarchical data structure;  
8 means for organizing a second set of the nonhierarchically organized classifiers (1:M)  
9 into a second hierarchical data structure according to a view established for a  
10 second client; and  
11 means for presenting data to the second client according the second hierarchical data  
12 structure.

1 (Original) The computer readable medium as recited in Claim 36, wherein the  
2 first hierarchical data structure is a file system directory tree structure.

1 (Original) The computer readable medium as recited in Claim 36, wherein the  
2 database includes N classifiers and N is a number greater than M.

1 (Original) The computer readable medium as recited in Claim 36, wherein the  
2 hierarchical data structure is a file system directory tree structure according to  
3 NFS.

1 (Previously Presented) A system for providing an interface to a database, wherein  
2 the database includes nonhierarchically organized classifiers(1:M) of data and  
3 data linked to at least one of the classifiers(1:M), wherein "M" is any positive  
4 number including 1, the system comprising:  
5 a data processing system having a memory coupled to at least one processor, wherein the  
6 memory comprises instructions for enabling the data processing system to:  
7 organize a set of the nonhierarchically organized classifiers(1:M) into a first  
8 hierarchical data structure according to a view established for a first client;  
9 present data to the first client according the first hierarchical data structure;  
10 organize a second set of the nonhierarchically organized classifiers (1:M) into a  
11 second hierarchical data structure according to a view established for a  
12 second client; and  
13 present data to the second client according the second hierarchical data structure.

1 (Original) The computer readable medium as recited in Claim 40, wherein the  
2 first hierarchical data structure is a file system directory tree structure.

1 (Original) The computer readable medium as recited in Claim 40, wherein the  
2 database includes N classifiers and N is a number greater than M.

1 (Original) The computer readable medium as recited in Claim 40, wherein the  
2 hierarchical data structure is a file system directory tree structure according to  
3 NFS.